

Алгоритм составления расписания

Проектная работа



21 февраля 2020 г.

МОсковский ПОлитех

Оглавление

[Алгоритм составления расписания 1](#_Toc34403104)

[Проверки при составлении первого варианта расписания 5](#_Toc34403105)

[Ограничения: 6](#_Toc34403106)

[Топология базы данных: 7](#_Toc34403107)

# Алгоритм составления расписания

**От преподавателя**

1. Выбираем самые загруженные группы по факторам:  
   Кол-во дисциплин;  
   Кол-во часов.
2. Находим Преподавателя с самым низким уровнем свобод;
3. Собираем пакет [Преподаватель с низшим уровнем свобод; Дисциплина; Самая загруженная Группа];  
   Далее собираем:
4. Возможная наиболее оптимальная аудитория (размер, проф. Пригодность, наличие проектора…);
5. Подбираем время. Получаем [Преподаватель с низшим уровнем свобод; Дисциплина; Самая загруженная Группа; Аудитория; Время];
6. // Возможные замечания  
   Если время плохое, то возвращаемся к шагу 3 и меняем группу (группу-дисциплину)  
   Если возможно совместить группы – совмещаем  
   Ставить рядом пары одного предмета  
   Если нет оптимальных решений – меняем препода

**От группы**

# Входные данные

Преподаватель:

1. Заполненная таблица приоритетов времени (по парам, например:  
   {[пн:9:00;10:40;12:10;17:50][вт:14:30;16:00;17:50][ср:….]})
2. Приоритет корпусов

Группа:

1. Приоритет корпусов

Дисциплина:

Учебная программа

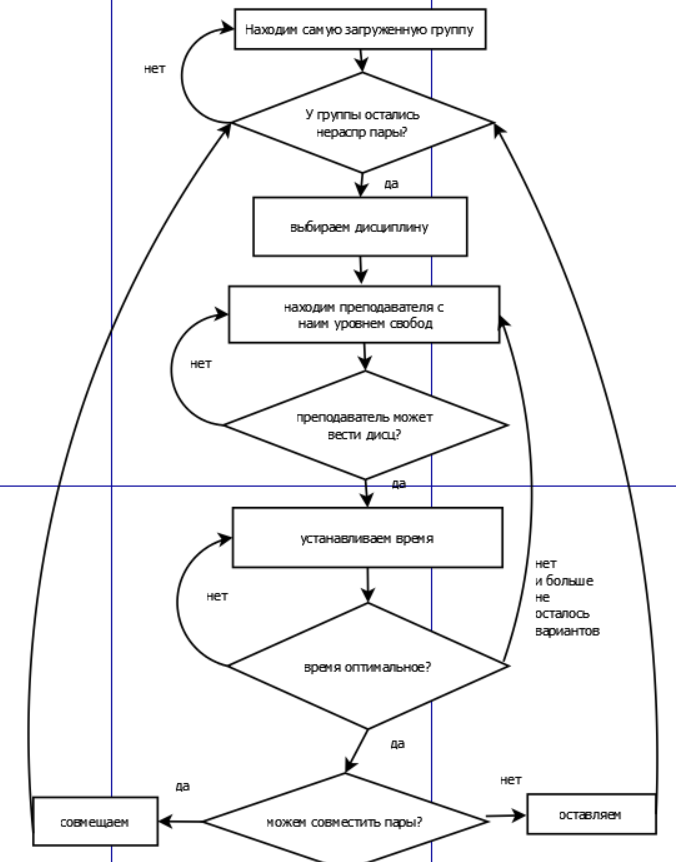
Требования

Аудитория:

* Требования

## Пояснения к алгоритму

1. Находим самую загруженную группу;
2. Если у группы не осталось нераспределенных пар, то возвращаемся к пункту 1. Если остались - к пункту 3;
3. Выбираем дисциплину;
4. Находим преподавателя с наименьшим количеством свобод;
5. Если преподаватель может вести пару (знает эту дисциплину, есть еще часы ((во избежание переработок)) ), то устанавливаем возможное время, если нет то к пункту 4; // если препод не нашелся
6. Время оптимальное? Если да, то пункт 7, если нет – 5. В случае отсутствия возможных вариантов выбираем нового преподавателя (пункт 4);
7. Если можем совместить пары, то совмещаем, нет – оставляем так.
8. Возвращаемся в п.2



# Проверки при составлении первого варианта расписания

Проверка преподавателя

1. Число пар меньше const N
2. Преподаватель может вести пару в это время
3. У него нет пар в этот день в других корпусах
4. \* Если пункт 3 невозможен, то между парами в разных корпусах необходимо окно
5. \* Если невозможно найти преподавателей, удовлетворяющих условию в пункте 1 увеличиваем значение N, тем самым мы сможем добиться максимально равномерной нагрузки на преподавателей.

Проверка аудитории

1. Аудитория свободна
2. Аудитория вмещает группу, но и не излишне велика для нее (оптимально на мой взгляд 125% от числа группы)
3. Аудитория находится в том же корпусе, что и предыдущая (если есть)
4. Аудитория имеет все требуемые условия для проведения занятия (проектор, компьютерный класс и т.п.)

Занятие

1. Если лекция – находим группу с того же потока if count\_student <= 60: объединяем пару в общую
2. \* Лекция перед практикой
3. \*\* Лекция и практика в один день
4. \*\*\* Если практика/лабораторная, то 1 преподаватель на 15 человек

Время

1. Преподаватель доступен
2. Пары одной дисциплины стоят подряд (но не более двух, если не физ-ра)

# Ограничения:

1. Вместимость аудитории
2. Наличие в ней нужного оборудования (компы с необходимым ПО, доска, станки и тд)
3. Нахождение в одном корпусе с предыдущей аудиторией в расписании того же дня (если это не первая пара)
4. Исключение "окон" между парами
5. Равномерность нагрузки студентов (и преподавателей желательно) в течение всего семестра и конкретного учебного дня
6. Допустимый максимум пар в день
7. Пары одной дисциплины в один день подряд (не больше 2х)
8. Самое раннее и позднее время начала пары
9. Пары лекций общие для групп одного направления (если не слишком много студентов)
10. Распределение дисциплины не на полный семестр
11. \* не ставить сложные пары в пн утром
12. \* не ставить пары вечером, а на следующий день утром
13. \*\*Не ставить общие пары 3-4 по счету, во избежание резкого упадка продуктивности студентов
14. \*\* вообще не ставить пары вечером
15. \*\*\*\* Стремиться создать 5 дневную рабочую неделю для студентов

# Топология базы данных:

**Дисциплина:**

* Название дисциплины
* Профессиональный тип (Обычный, Компьютерный, Дизайн, Лабораторная, Мастерская)
* Нужен проектор
* Нужна большая доска
* Преподаватели (отношение многие ко многим)

**Преподаватель:**

* Имя
* Фамилия
* Отчество
* Дисциплины (отношение многие ко многим)
* Ограничения
  + - { 'buildings\_priority': [] - приоритет корпусов,
    - 'day\_constraints': {'день недели': [7 булевых значений по каждой паре]}}
* Общее количество рабочих часов

**Направление подготовки:**

* Код подготовки
* Название подготовки
* Тип (Бакалавриат, Специалитет, Магистратура)
* Ограничения: (пока что только по парам как у преподавателей day\_constraints)

**Поток:**

* Название

**Группа:**

* Код (код группы)
* Количество студентов
* Направление подготовки (внешний ключ на Направление подготовки)
* Поток (внешний ключ на поток)
* Ограничения группы (пока что только по парам как у преподавателей day\_constraints)

**Образовательная программа:**

* Дисциплина (внешний ключ)
* Группа (внешний ключ)
* Тип (Лекция, Практика, Лаб. работа)
* Количество часов
* Ограничения (продумать)

**Аудитория:**

* Вместительность
* Код (номер аудитории)
* Здание (где расположена)
* Проф тип (также как у дисциплины)
* Есть ли проектор
* Есть ли большая доска

**Пара (уже составленное расписание):**

* Дисциплина (внешний ключ на образовательную программу)
* Группа (внешний ключ)
* Преподаватель (внешний ключ)
* Аудитория (внешний ключ)
* Номер пары
* День недели

# Оптимизация расписания

**Цель**: составить максимально оптимальное расписание путем увеличения коэффициента О(оптимальность) -> к max до максимально крупного значения